
ATLAS Distributed Computing Experience and Performance During the LHC Run-2

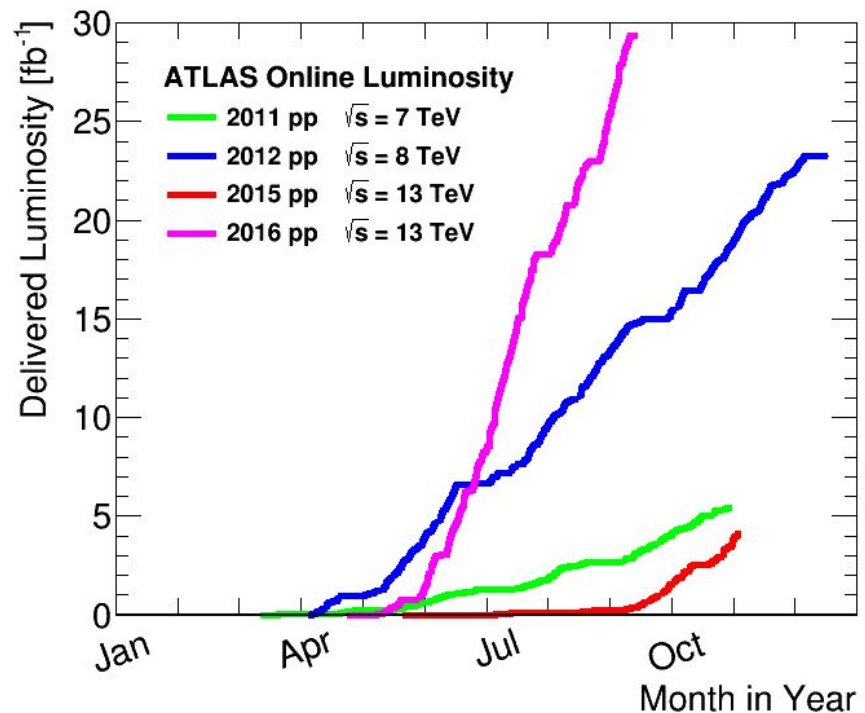
Andrej Filipčič on behalf of
the ATLAS Collaboration

N.b.: this talk will focus on an ATLAS Distributed Computing overview, for more details check the 40+ other contributions from the ADC community

Run-2 experience

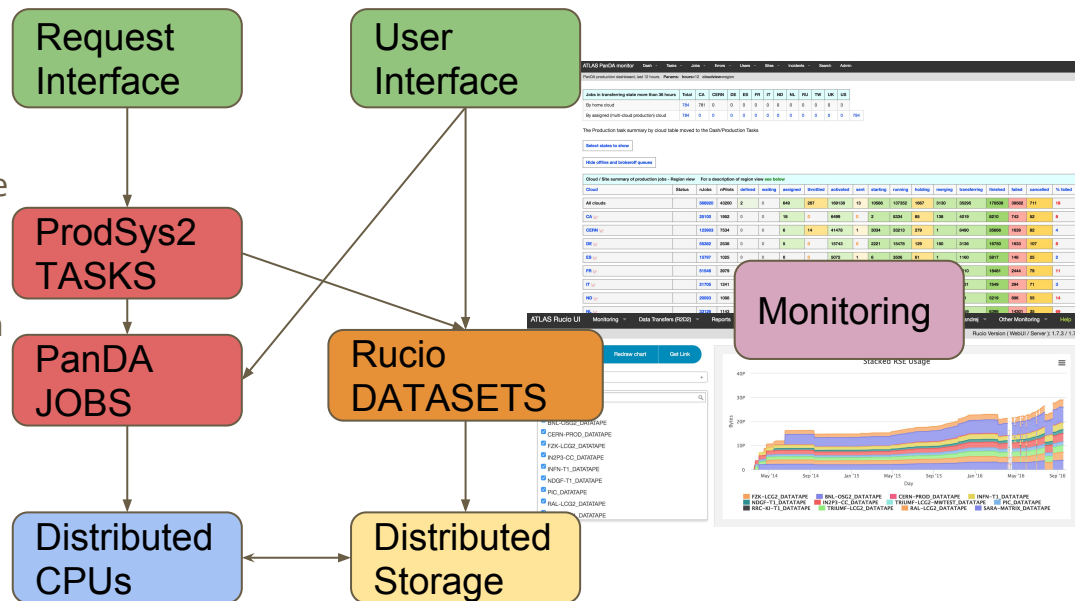
- LHC is performing really well:
 - Duty cycle ~80%
 - Higher luminosity in 2016
 - 50% more data delivered than expected
- Allocated computing resources are not sufficient to cope with the data-taking load but...
- Sites provide more CPU power than requested
- New production framework developed during LS1 performs really well and efficiently exploit the resources

ATLAS computing works extremely well and provides the results on time for conferences



New production and data management system

- Workflow Management System
 - PanDA/JEDI
 - Dynamic resources, jobs
 - Analysis and production use the same infrastructure
 - ProdSys2
 - Workflow organization relies on input transformation
 - Any kind of workflow is quickly implemented
- Rucio
 - Optimized and scalable data management
 - Transfer latencies are minimized



- 140. PanDA for ATLAS distributed computing in the next decade
- 124. The ATLAS Production System Evolution. New Data Processing and Analysis Paradigm for the LHC Run2 and High-Luminosity
- 147. Experiences with the new ATLAS Distributed Data Management System

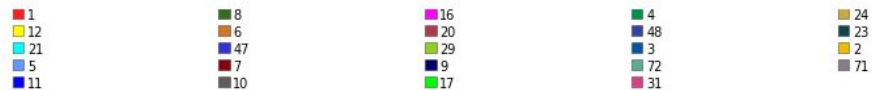
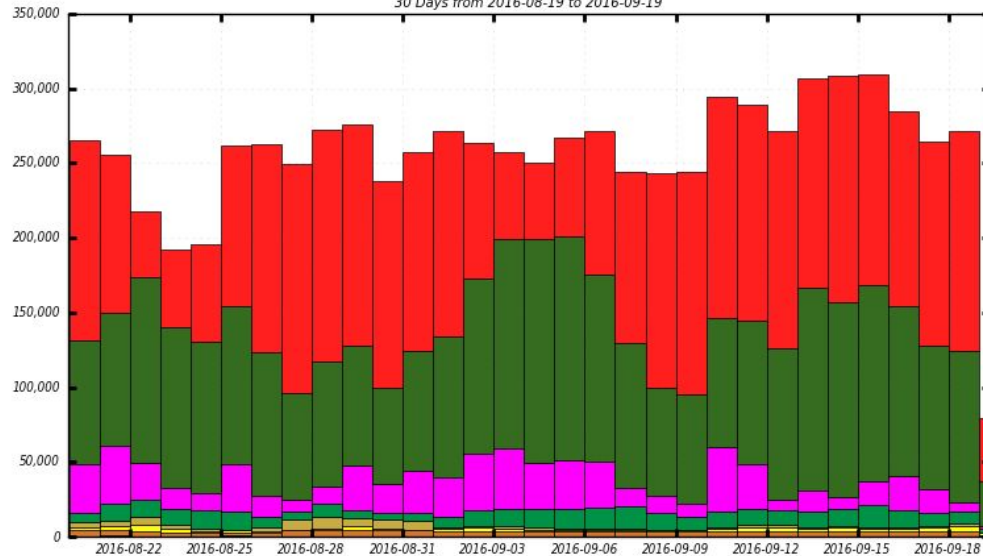
Other changes since Run-1

- WORLD cloud
 - MONARC model is gone
 - Every stable site can store primary data (nucleus)
 - Every site well connected to nucleus can process data (satellite)
 - All associations are fully dynamic at the task and job brokering level
 - See 151. [ATLAS World-cloud and networking in PanDA](#)
- Auto-tuning of jobs:
 - Memory and walltime of jobs are measured for first 10 jobs of a task and set for the rest
 - Retries of failed jobs have increased memory or walltime if that was the reason for failure
- Task completion
 - Requests and tasks are monitored for progress
 - Almost completed tasks or tasks with a close deadline have proactive push to complete the remaining jobs

Flexible and dynamic resource allocation



Slots of Running Jobs
30 Days from 2016-08-19 to 2016-09-19



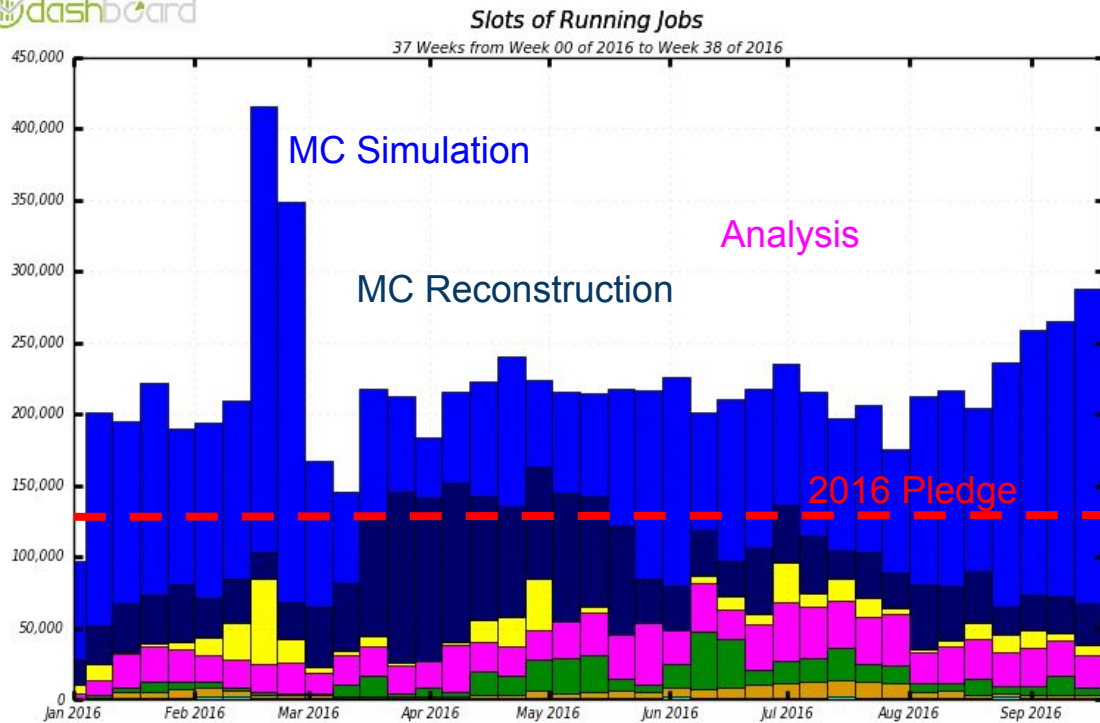
Maximum: 309,861 , Minimum: 0.00 , Average: 248,131 , Current: 79,117

- Single and multicore jobs are effectively using the resources
 - Dynamic allocations
- High-memory jobs
 - Activities such as upgrade studies, Heavy-Ion processing need more memory
 - Custom high-memory queues or dynamic site queues can provide higher memory on demand at the expense of

unused cores

CPU usage

- Using much more CPU than pledged
- Overpledged - using the additional CPUs on WLCG sites (50%)
- Opportunistic - sites that are not part of WLCG (15%)
 - HPCs
 - Academic and commercial clouds
 - Shared facilities
 - ATLAS@Home
- Significant I/O stress:
 - Higher pile up - MC reconstruction
 - Longer I/O intensive campaigns



■ MC Simulation
■ TO Processing

■ MC Reconstruction
■ Others

■ Data Processing
■ unknown

■ Analysis

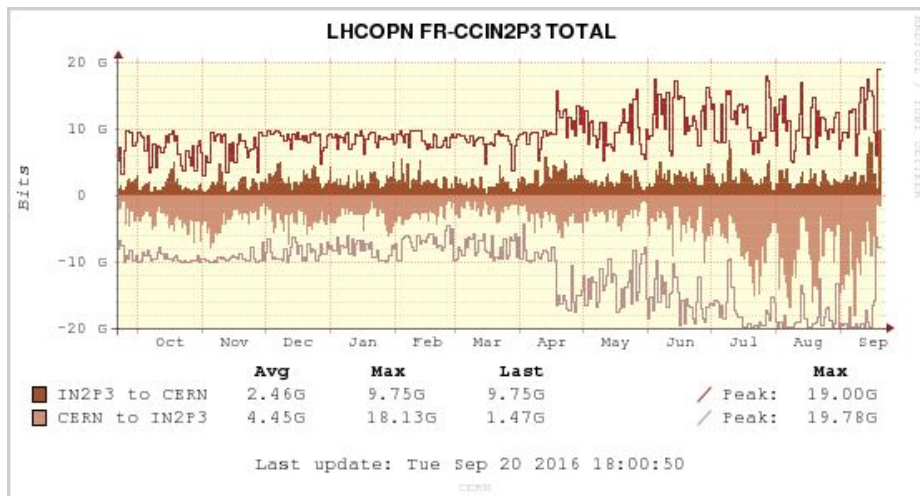
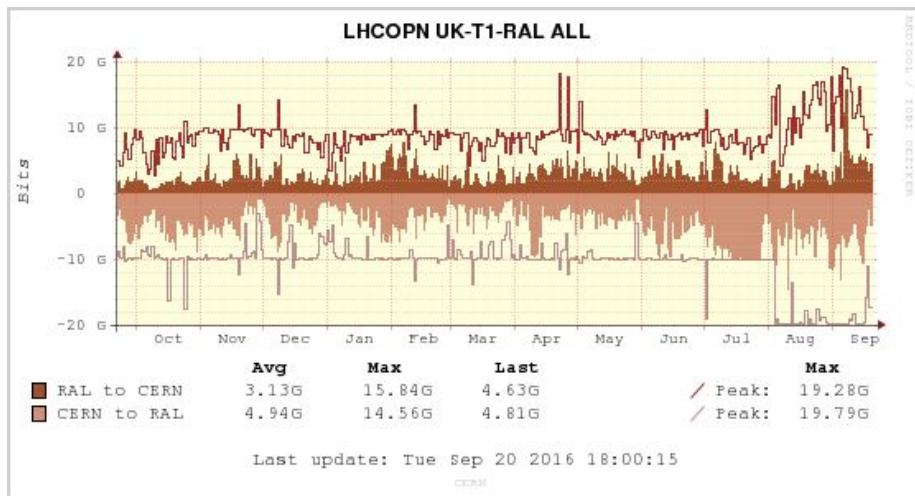
■ Group Production

Maximum: 415,666 , Minimum: 0.00 , Average: 208,699 , Current: 11,302

Network is not infinite

- LHCOPN pushed to its limit
- Secondary link, usually used for resiliency, actively used now

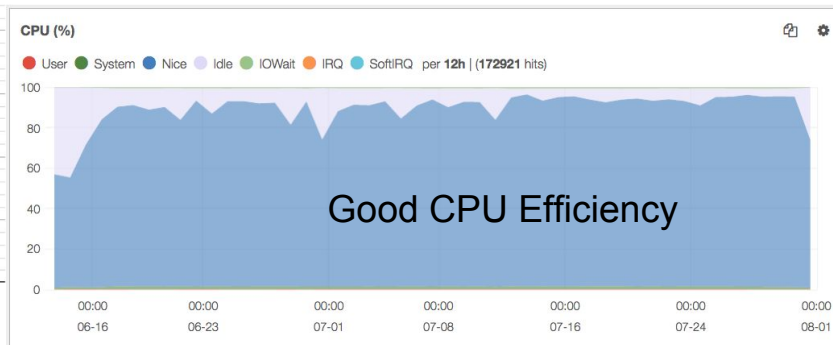
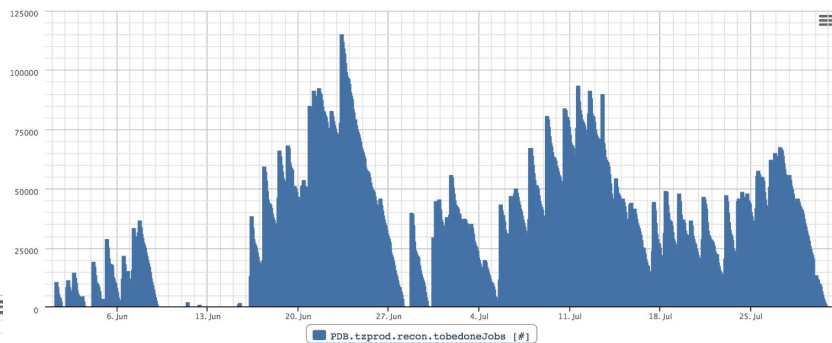
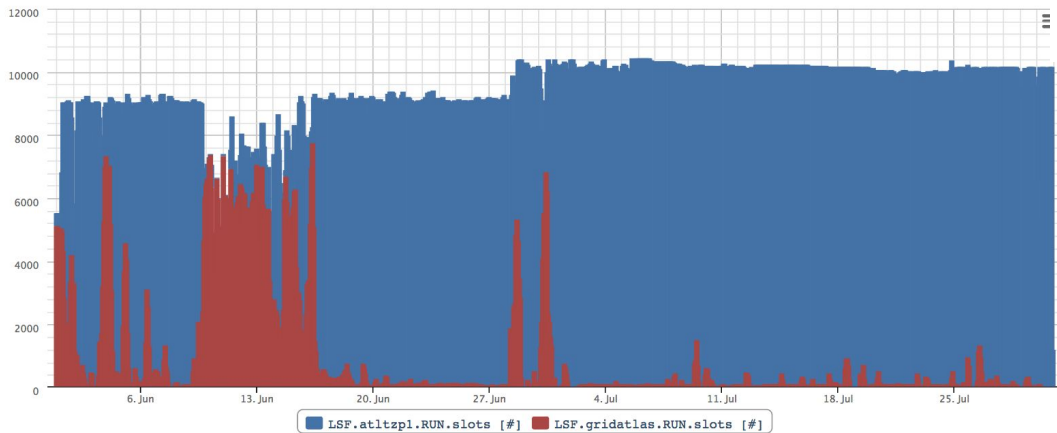
more in
91. [Networks in ATLAS](#)



Tier-0 processing

- Data taking pushed the infrastructure to the limit
- Powerful worker nodes:
 - SDD, 4GB memory/core, Hyper Threading switched off
 - ~10k cores
- Tier-0 data processing jobs
- Grid jobs running on Tier-0 when idle

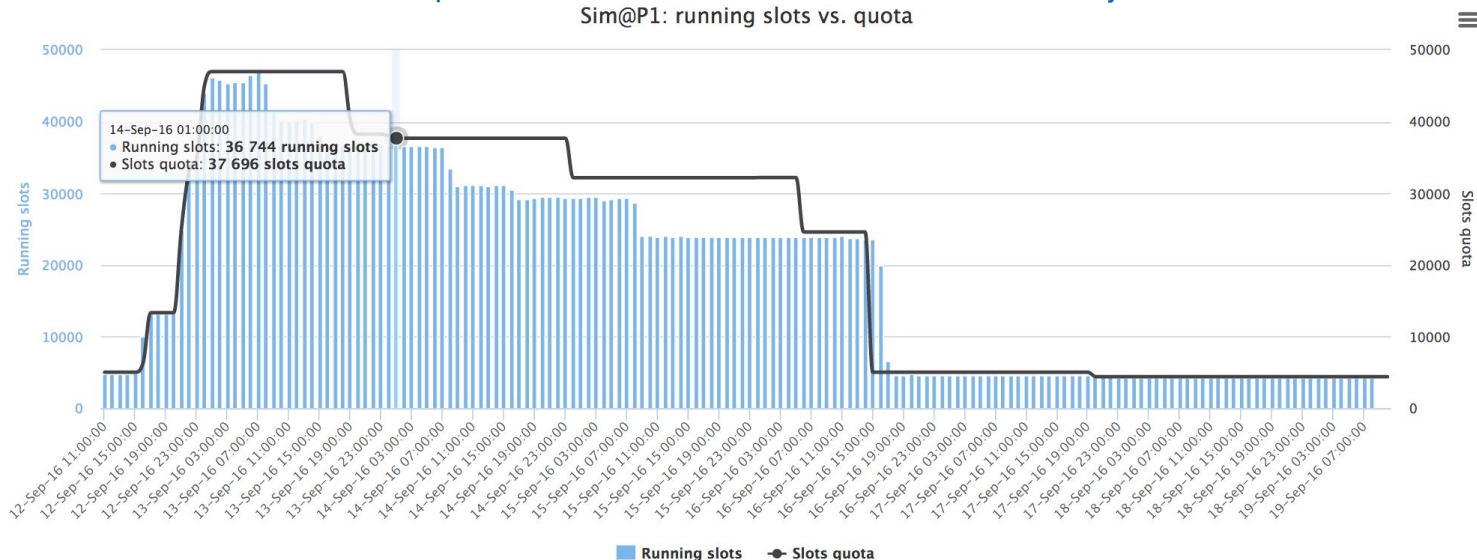
- Waiting data processing jobs
- Largest runs (~40h) processed in 4 days



Exploiting HighLevelTrigger farm

- HLT cores are effectively used by distributed production when not in data taking (some cores are reserved for trigger development)
- Quick job assignment is essential
- See 143. [Evolution and experience with the ATLAS Simulation at Point1 Project](#)

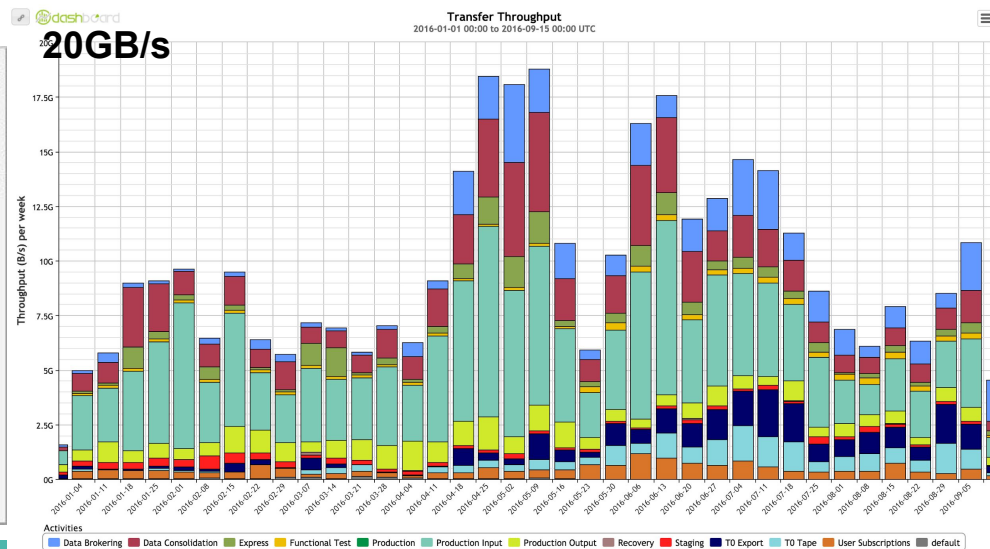
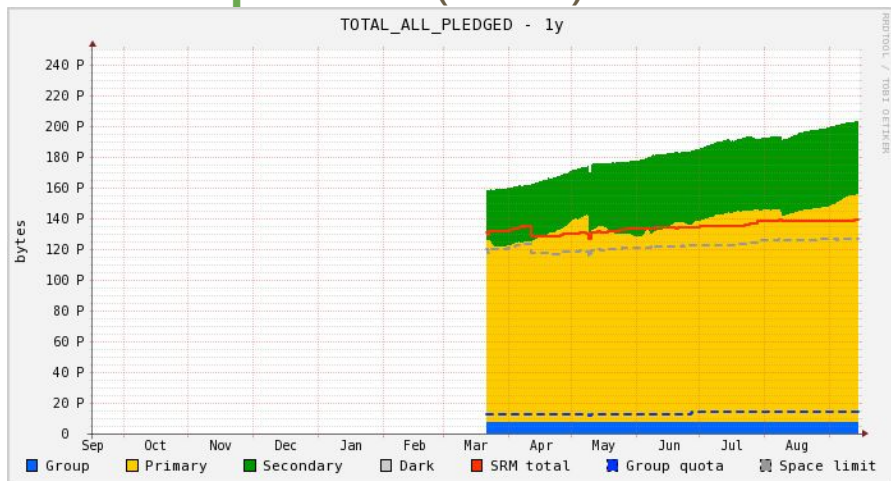
Sim@P1: running slots vs. quota



Data management

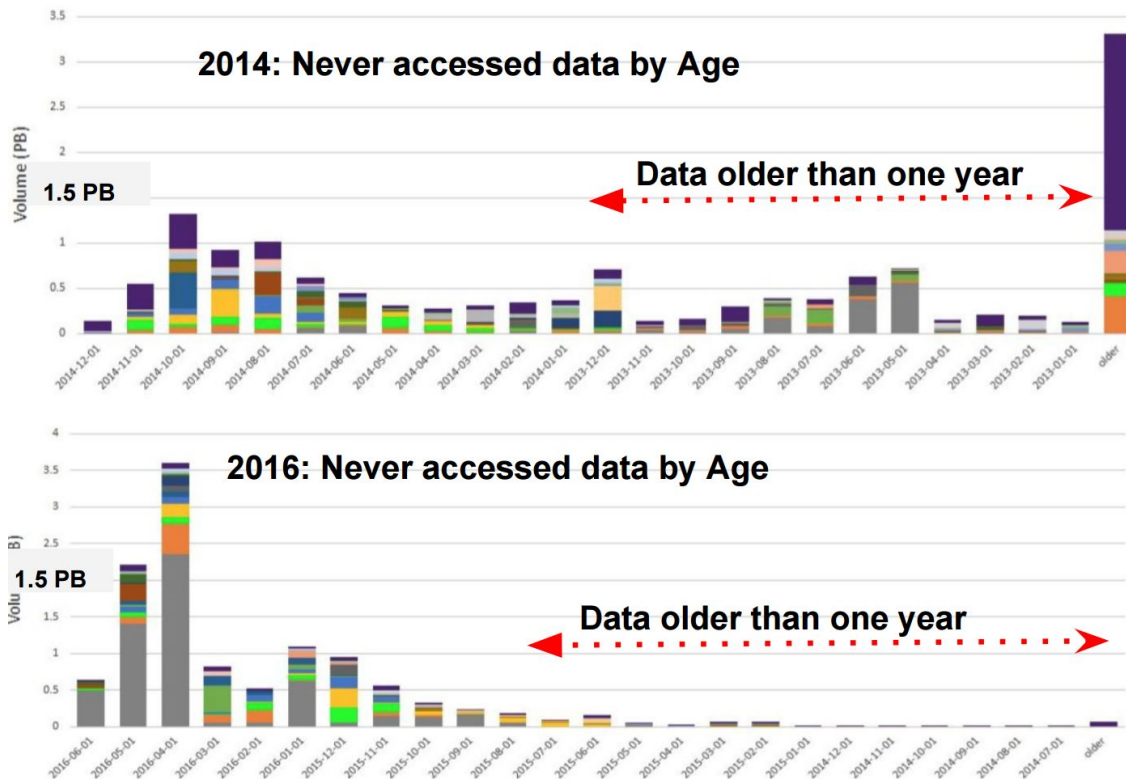
- Data Volume exceeds 200PB on disk and tape
 - 1B files, 100k datasets
- **Primary data** is partially **replicated** (~30%)

- Data transfers reaching 20GB/s weekly and 40GB/s daily average
- More than 50 files/s transferred
- Largest activity - **input transfers**



Data persistency - the lifetime model

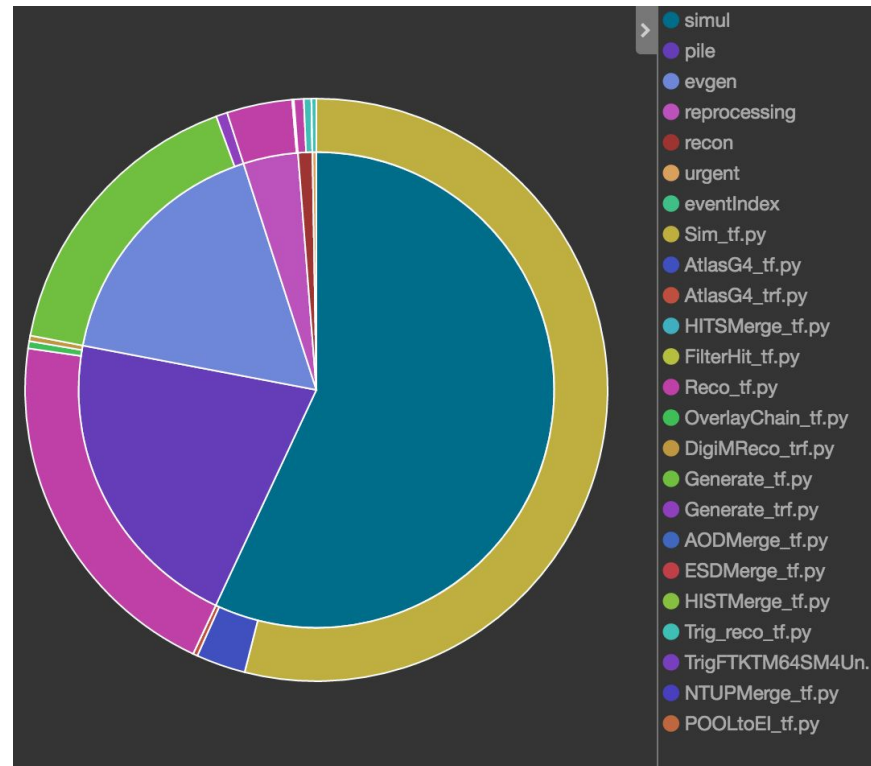
- There is too much data to keep on storage permanently
- Each data type is set a finite lifetime:
 - Analysis inputs (DAOD) - 6 months, fast turnaround
 - Monte-Carlo simulations - 2-3 years, expensive to regenerate
 - RAW data - unique and precious, infinite lifetime
- Frequently used data - lifetime extension
- Monthly cleanup procedure for expired datasets
 - Approval of exceptions
 - Permanent automated deletion of expired data



Production activities in 2016

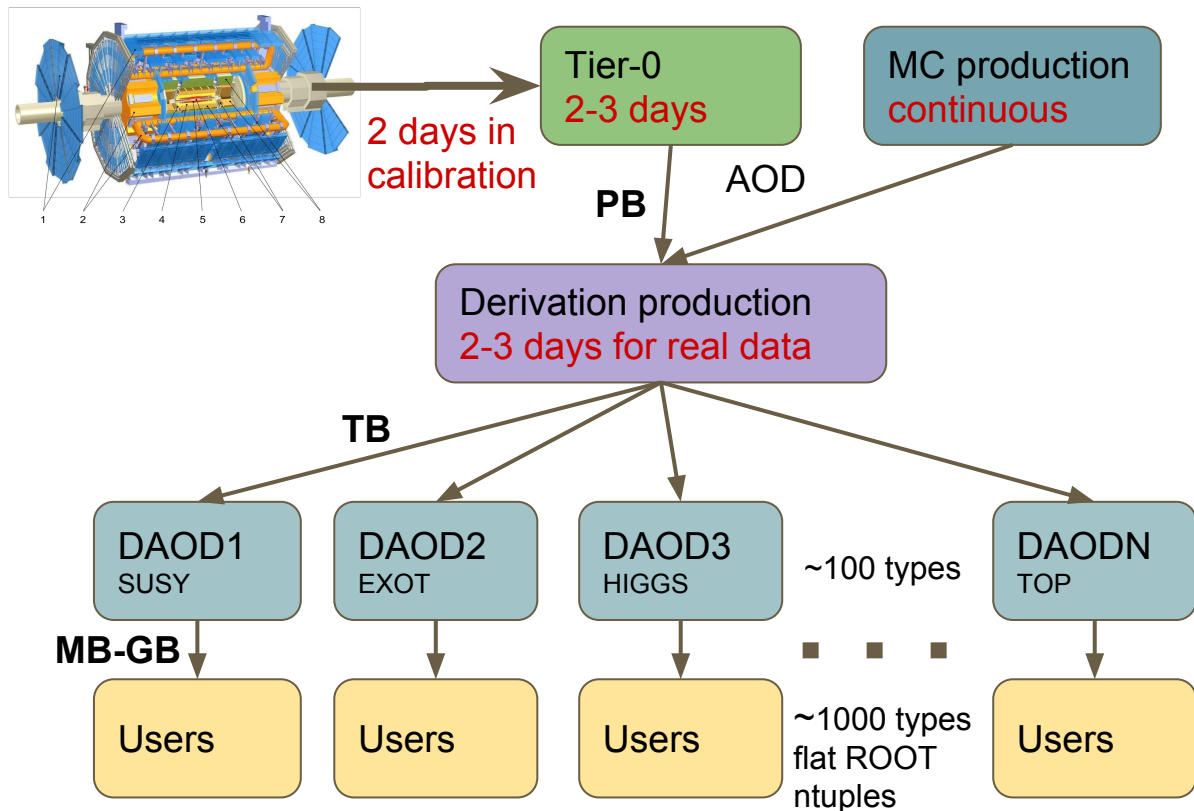
- MC15c campaign (94%)
 - Produce ~5B of full simulation events
 - Reconstruction with 2016 data-taking conditions
- Processing of 2015 Heavy-Ion data and partial pp reprocessing (4%)
- Several derivation production campaigns during 2016
 - Full processing of MC15c, 2015 and 2016 data
 - 2-3 weeks for a full campaign

CPU usage by processing type



Derivations production

- Centrally managed production of analysis specific DAOD datasets (reduced data format from main AOD format)
- Real data:
 - Available ~1 week after data taking
- Several campaigns with improved analysis code on data and MC
 - 2-3 weeks to process



Upcoming features

- Global fair-shares
 - Limit the cpu slots per activity
 - Boost activity when requested
 - Hierarchical implementation - sub-activities within activity have partial shares
 - When in production, analysis and production will run within the same batch account - boosting analysis before conferences
- Machine learning studies and analytics
 - All the monitoring records are stored in Elasticsearch for detailed analysis
 - Optimizing the request completion time by machine learning algorithms
 - Using Network Weather Service to speed-up processing and reduce the network usage
 - Job error analysis algorithms to automatically handle service or site failures
 - 131. [Using machine learning algorithms to forecast network and system load metrics for ATLAS Distributed Computing](#)
- Event service
 - See 92. [Production Experience with the ATLAS Event Service](#)

Conclusions

- Big efforts to evolve and partially redesign the production system has paid off
 - Production and data management system is robust, performant and much lighter operationally than the system used during Run-1
 - Can easily cope with higher Run-2 LHC performance than expected
- There are presently no scaling issues. Each subsystem is tested to sustain 5-10 times larger load
 - Still, it might not be sufficient for high-luminosity LHC (Run-4)
- Processing power, storage capacities and network used to its limits
 - Careful planning of cpu and storage usage for campaigns
 - Cleaning unused data
 - Optimizing network usage to avoid saturation